50p

mfs N64 16112

code 1

CR55654

UNPUBLISHED PRELIMINARY DATA

OTS PRICE

XEROX $ 4.60ph

MICROFILM $ 1.70mf.

1485494

UNIVERSITY OF MARYLAND U., College Park

COMPUTER SCIENCE CENTER

COLLEGE PARK, MARYLAND

Technical Report TR-63-4
NsG-398

November 1963

t: The Use of Digital Computers to Determine Definitions for Abstract Groups

NsG 398

by

James W. Snively, Jr.
N.A.S.A. - Trainee
Computer Science Center

Nov. 1963 50p refs

NASA Grant NsG 398

(NASA CR-55654; TR-63-4) OTS: $ . . .

cite only

ACKNOWLEDGEMENTS

The author of this paper is grateful to Dr. A. Sinkov and Dr. S. Kuroda for their guidance and encouragement in the preparation of this paper, to the Computer Science Center of the University of Maryland for the use of their facilities, and to N.A.S.A. for support for the research.

The author's graduate studies are directly supported by N.A.S.A. under a N.A.S.A. Traineeship stipend granted to the University of Maryland. The computer time involved was supported by N.A.S.A. grant NsG-398 awarded to the Computer Science Center.

ABSTRACT

16112

This paper describes the work done by the author and others on digital computer programs for automatic enumeration of cosets. It is followed by a brief description of some of the work done on the finite "Burnside" groups especially by computer enumerations. A definition for the Burnside group $B_{3,4}$ of exponent 3 with 4 generators involving only 35 words is given. AUTHOR

CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

CHAPTER I

BASIC PROBLEM

If it is known that a group, G, is generated by a set of generators: $S_1, S_2, \ldots, S_k$ and that these generators satisfy a set of relations of the form:

$$f_i(S_1, S_2, \ldots, S_k) = E, \quad (i = 1, 2, \ldots, n) \tag{1}$$

then a natural question which arises is: What is the order of the group thus defined? An equivalent way of expressing this question is to ask how many of the combinations and permutations of the generators are distinct when subjected to the constraints implied by the relations, f_i, and the definition of an abstract group. For all but the most trivial of groups the direct approach clearly involves too much manipulation to be practical.

The most obvious technique to reduce the magnitude of the problem is to choose as large a subgroup, H, of G as possible -- the order of H being known from the set of defining relations -- and enumerate all of the left[1] cosets of that subgroup. For example, suppose one wished to determine the

---

[1]The choice of left or right cosets is immaterial to the process. The reason for choosing left cosets is explained in the paragraph of Chapter II.

order of the group, $(8,7|2,3)$[2], defined by the relations:

$$A^8 = B^7 = (AB)^2 = (A^{-1}B)^3 = E \qquad (2)$$

There are several choices for a subgroup. One may use the group generated by B, denoted $\{B\}$, which is the cyclic group of order 7, or $\{A\}$, the cyclic group of order 8. However, a still better choice is $K \equiv \{A^2, A^{-1}B\}$. A^2 is of period 4, $A^{-1}B$ is of period 3, and their product, $A^2 \cdot A^{-1}B = AB$, is of period 2. These generators satisfy the relations (2,3,4):

$$R^2 = S^3 = (RS)^4 = E \qquad (3)$$

which define the symmetric group on four elements, S_4, of order 24. This is easily seen by setting $B^{-1}A = S$ and $AB = R$. Then $RS = AB \cdot B^{-1}A = A^2$. The order of $(8,7|2,3)$ is known to be 10,752; therefore, using the subgroups specified only 1536, 1344, and 448 cosets would have to be enumerated respectively according as the basic subgroup is $\{B\}$, $\{A\}$ or K.

It is well known from elementary group theory that distinct cosets of the subgroup, H, have empty intersections; that any given coset, aH, is independent of the choice of the

---

[2]This group, which was first described by Dr. A. Sinkov; "On the group-Defining Relations (2,3,7;p)" Annals of <u>Mathematics</u> XXXVIII (1937), pp 580-582 and was later given an irreducible definition, has been widely used as a test problem for computer enumeration problems and thus for a comparison of the various techniques.

representative, a, that is $b \in aH \Rightarrow aH = bH$; and that the union of the collection of all cosets of H is the entire group, G:

$$G = a_1 H \cup a_2 H \cup \ldots \cup a_m H \qquad (4)$$

where $a_1 \in H$ and $a_i H \cap a_j H = \emptyset$, $i \neq j$

Thus if in an enumeration of G by cosets of H, m cosets were defined, the order of G would be m times the order of H.

CHAPTER II

SYSTEMATIC ENUMERATION

In 1936 Todd and Coxeter[3] presented a mechanical method for accomplishing this type of enumeration. Except for changes involving the choice of considering the elements of the group, G, as right multipliers, the process to be described below is that of Todd and Coxeter. The choice of right multipliers (left cosets) produces tables which are read in the conventional manner.

Consider a typical member of the set of relations (1). It would have the form:

$$R_1 R_2 \ldots R_\ell = E \tag{5}$$

where each R_i is one of the generators or the inverse of one of the generators and generators are repeated if necessary. Such a relation is called a <u>word</u> of ℓ <u>letters</u>. For example, if the commutator of generators A and B were to have its period specified as 2, (5) would have the form:

$$A^{-1}B^{-1}ABA^{-1}B^{-1}AB = E \tag{6}$$

In the process about to be described, there is a coset multiplication table for each word in the set of defining

---

[3] J. A. Todd and H. S. M. Coxeter. "A Practical Method for Enumerating Cosets of a Finite Abstract Group," <u>Proceedings of the Edinburgh Mathematical Society</u>, V, Series II (1936), pp. 26-34.

relations, (1). Each table has one column more than the number of letters in the corresponding word. The generators forming the word are placed at the head of the table between the various columns as in the following example: S_4 defined by the relations (3).

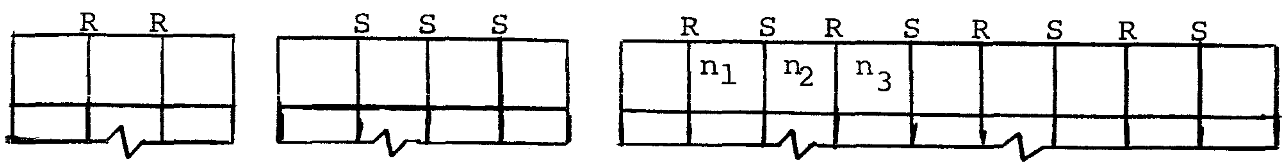

Fig. 1 -- Diagram of typical enumeration table

In Figure 1 the numbers n_1, n_2, and n_3 represent cosets. The entries are to be interpreted as $n_1 \cdot S = n_2$ and $n_2 \cdot R = n_3$ and, when reading backwards, as $n_3 \cdot R^{-1} = n_2$ and $n_2 \cdot S^{-1} = n_1$. The constraints implied by the defining relations (1) are applied by requiring the coset numbers at both ends of a row to be identical.

The process is initiated by inserting in the tables the choice for coset 1, namely the subgroup H. In the example cited above, a reasonable choice for H is the subgroup generated by S, the cyclic group of order 3. Hence we denote H = S = 1. It follows that $1 \cdot S = 1$ since any element H is of the form S and S · S is also of the form S . Thus the above result is entered into the tables in every possible position as follows:

R R

1		1

S S S

1	1	1	1

R S R S R S R S

1							1	1

Fig. 2.--Initial stage of enumeration

Note that at each stage of the process every table should contain a row that begins and ends with each of the cosets already defined. However, if any of the words may be put in the form

$$(R_1R_2\ldots R_\ell)^\lambda = E \qquad (7)$$

then there is a λ-fold symmetry. Repetition of rows beginning with λ-1 of the cosets will only result in repetition of the λ-th row with the entries rotated cyclicly. These repetitions are superfluous and therefore may be omitted.

At each stage in the process when it has been determined that no more information can be inserted in the tables, the next coset is defined to fill an empty space in the tables. In our example such a definition could be $1 \cdot R = 2$. Then the tables would be filled in as follows:

R R

1	2	1

S S S

1	1	1	1
2			2

R S R S R S R S

1	2					2	1	1

Fig. 3.--Second stage of enumeration

In Figure 3 it is deduced that $2 \cdot R = 1$. Algebraically this deduction is a consequence of the relation $R^2 = E$. Coset 2 is of the form $S^\alpha R$. Thus $2 \cdot R$ is of the form $S^\alpha \cdot R^2 = S^\alpha E = S^\alpha = 1$. As the process continues, more instances of deduced information will occur. The advantage of this mechanical procedure is that the algebraic consequences of the defining relations are handled automatically, thus eliminating the necessity for abstract manipulation.

Also note that $(RS)^4 = E$ is a relation of the form (7) as in fact are the other two. Therefore, a row beginning and ending with coset 2 is superfluous and hence was omitted in Figure 3.

The next pair of definitions might be $2 \cdot S = 3$ followed by $3 \cdot S = 4$ from which it is deduced that $4 \cdot S = 2$. At this point, the tables appear as follows:

	R	R
1	2	1
3		3
4		4

	S	S	S
1	1	1	1
2	3	4	2

	R	S	R	S	R	S	R	S
1	2	3			4	2	1	1
4							3	4

Fig. 4.--Later stage of enumeration

From this point one might proceed as follows. Define $3 \cdot R = 5$ thus deducing that $5 \cdot R = 3$. Then define $4 \cdot R = 6$ deducing that $6 \cdot R = 4$ and $5 \cdot S = 6$. At this point the tables appear as follows:

	R		R
	1	2	1
	3	5	3
	4	6	4

	S		S		S	
	1	1	1	1
	2	3	4	2
	5	6		5

	R		S		R		S		R		S		R		S	
	1*	2	3*	5	6*	4	2*	1	1
	4*	6					5*	3	4

Fig. 5.--Still later stage of enumeration

Note that the starred coset numbers are in symmetrical positions and hence only two rows need to be carried in the chart corresponding to the relation $(RS)^4 = E$.

The next pair of definitions might then be $6 \cdot S = 7$ which leads to $7 \cdot S = 5$ and finally $7 \cdot R = 8$ from which the results $8 \cdot R = 7$ and $8 \cdot S = 8$ are easily found. At this point the tables have "closed up" and the process is complete. The resulting tables appear below:

	R		R
	1	2	1
	3	5	3
	4	6	4
	7	8	7

	S		S		S	
	1	1	1	1
	2	3	4	2
	5	6	7	5
	8	8	8	8

	R		S		R		S		R		S		R		S	
	1	2	3	5	6	4	2	1	1
	4	6	7	8	8	7	5	3	4

Fig. 6.--Completed enumeration

There were 8 cosets defined and since H is of order 3, the order of the group (2,3,4) is 24. The coset multiplication table may be summarized in the following diagram:

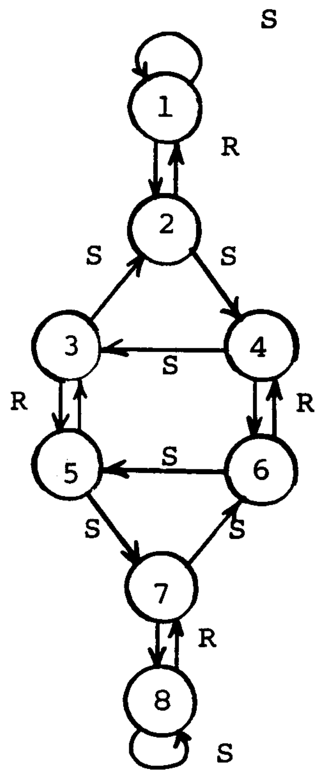

Fig. 7.--Coset diagram for (2,3,4)

Any path from 1 to n yields an expression for an element of coset n. For example, the element $SRSRS^2R$ in coset 8 corresponds to the path 1-1-2-4-6-5-7-8 on Figure 7.

In the process, it may occur that two different coset numbers are discovered to actually represent the same coset. When such a coincidence occurs, the larger number is replaced throughout the tables by the smaller and the process continues. When working by hand it may be convenient to use the larger number as that of the next coset defined.

As an example of a simple enumeration which involves a coincidence consider Klein's simple group of order 168 defined by the relations (2,3,7;4):

$$P^2 = Q^3 = (QP)^7 = (Q^{-1}P^{-1}QP)^4 = E \qquad (8)$$

Let $PQ = R$ and $Q^{-1} = S$. Then we have $P = PQ \cdot Q^{-1} = RS$ and $Q^{-1}P^{-1}QP = (PQ)^{-1}\, Q \cdot PQ \cdot Q^{-1} = R^{-1}S^{-1}RS$. Thus relations (8) imply

$$R^7 = S^3 = (RS)^2 = (R^{-1}S^{-1}RS)^4 = E \qquad (9)$$

We now define the basic subgroup as $\{R\}$ so that $1 \cdot R = 1$ and proceed as indicated in the following table:

TABLE 1: DEFINITIONS FOR (2,3,7;4)

Definition	Implied Consequences		
$1 \cdot S = 2$			
$2 \cdot S = 3$	$3 \cdot S = 1$	$2 \cdot R = 3$	
$3 \cdot R = 4$			
$4 \cdot S = 5$	$5 \cdot R = 2$		
$5 \cdot R^{-1} = 6$			
$6 \cdot S^{-1} = 7$			
$7 \cdot R = 8$			
$8 \cdot S = 9$			
$9 \cdot R^{-1} = 10$			
$10 \cdot S^{-1} = 11$			
$11 \cdot R = 12$	$12 \cdot S = 4$	$5 \cdot S = 12$	$12 \cdot R = 7$
$4 \cdot R = 13$	$13 \cdot S = 11$	$10 \cdot S = 13$	
$13 \cdot R = 14$	$14 \cdot R = 6$		
$11 \cdot R^{-1} = 15$	$9 \cdot S = 15$	$15 \cdot S = 8$	
$9 \cdot R = 16$	$16 \cdot S = 7$	$6 \cdot S = 16$	
$14 \cdot S^{-1} = 17$	$16 \cdot R = 17$		
$17 \cdot R = 18$			
$18 \cdot S = 19$			
$19 \cdot R^{-1} = 20$			
$20 \cdot S^{-1} = 21$			
$21 \cdot R = 22$			
$22 \cdot S = 23$			
$23 \cdot R^{-1} = 24$			

At this point the tables appear as in figure 8:

	R	R	R	R	R	R	R	
1	1	1	1	1	1	1	1	
2	3	4	13	14	6	5	2	
7	8			15	11	12	7	
9	16	17	18			10	9	
19						20	19	13
21	22						21	
23						24	23	

	S	S	S	
1	2	3	1	
4	5	12	4	
6	16	7	6	
8	9	15	8	
10	13	11	10	7
14		17	14	
18	19		18	
20		21	20	
22	23		22	
24			24	

	R	S	R	S
1	1	2	3	1
3	4	5	2	3
4	13	11	12	4
6	5	12	7	6
7	8	9	16	7
8			15	8
10	9	15	11	10
13	14		10	13
14	6	16	17	14
17	18	19		17
18				18
20	19		21	20
21	22	23		21
22				22
24	23			24

	R^{-1}	S^{-1}	R	S	R^{-1}	S^{-1}	R	S	R^{-1}	S^{-1}	R	S	R^{-1}	S^{-1}	R	S
1	1	3	4	5	6	7	8	9	10	11	12	4	3	2	3	1
2	5	4	13	11	15	9	16	7	12	5	2	3	2	1	1	2
6	14	17	18	19	20	21	22	23	24		10	13	4	12	7	6
8	7	16	17	14	13	10	9	15							15	8
10	9	8											8	15	11	10
12	11	13	14									17	16	6	5	12
16	9	8												14	6	16
18	17															18
20															21	20
21																21
22	21															22
24																24

Fig. 8 -- (2,3,7;4) before the first coincidence

If the next definition made is: $24 \cdot S^{-1} = 25$ we deduce from $(R^{-1}S^{-1}RS)^4 = E$ that $25 \cdot R = 10$ and then from $(RS)^2 = E$ that $14 \cdot S = 25$. At this point we have coset 25 in two previously distinct rows in the table corresponding to $S^3 = E$. Comparing the rows we discover that $17 \equiv 24$, i.e. that the cosets represented by these two numbers are really the same. Next by comparing the rows containing 17 and 24 in the tables corresponding to $R^7 = E$ we deduce that $18 \equiv 23$. A further search of the tables quickly indicates that no further consequential coincidences result. Thus we replace 24 by 17, 23 by 18, and 25 (the only coset defined after 23 and 24) by 23 (the lowest vacated coset number) and eliminate the rows which are duplicated. The resulting tables are then shown in figure 9:

1	1	1	1	1	1	1	1
2	3	4	13	14	6	5	2
7	8			15	11	12	7
9	16	17	18		23	10	9
19						20	19
21	22						21

1	2	3	1
4	5	12	4
6	16	7	6
8	9	15	8
10	13	11	10
14	23	17	14
18	19	22	18
20		21	20

1	1	2	3	1
3	4	5	2	3
4	13	11	12	4
6	5	12	7	6
7	8	9	16	7
8			15	14
10	9	15	11	10
13	14	23	10	13
14	6	16	17	14
17	18	19		17
20	19		21	20
21	22	18		21
22				22

1	1	3	4	5	6	7	8	9	10	11	12	4	3	2	3	1
2	5	4	13	11	15	9	16	7	12	5	2	3	2	1	1	2
6	14	17	18	19	20	21	22	18	17	23	10	13	4	12	7	6
8	7	16	17	14	13	10	9	15							15	8
10	9	8											8	15	11	10
12	11	13	14									17	16	6	5	12
16	9	8												14	6	16
20															21	20
21																21
22	21															22

Fig. 9--(2,3,7;4) after processing the first coincidence

We may then proceed as before to finish the enumeration. It turns out with one scheme of definitions (that used by the author's computer enumeration program) that 27 cosets will have to be defined and then a coincidence will reduce the number to 24 at which point the tables close-up. Thus the group (2,3,7;4) is of order $7 \cdot 24 = 168$ as expected.

The order of defining new cosets is completely immaterial to the success of an enumeration. However, by a judicial sequence of definitions, the number of coincidences may be minimized. Experience seems to indicate that many groups cannot be enumerated without the occurrence of coincidences. As interesting examples of this I refer to John Leech's recent paper[4] where he cites private correspondence with Todd suggesting that two groups: Klein's simple group of order 168 defined by:

$$\begin{gathered} B^7 = (AB)^2 = (A^{-1}B)^3 = (A^2B^2)^4 = E \\ \text{or} \\ B^7 = (AB)^2 = (A^{-1}B)^3 = (A^3B^4)^3 = E \end{gathered} \tag{10}$$

and the previously cited group, $(8,7|2,3)$, defined by (2), when enumerated as cosets of $\{B\}$ and $\{A^2, A^{-1}B\}$ respectively, cannot be enumerated without the occurrence of coincidences.

[4]John Leech. "Coset Enumeration on Digital Computers," Proceedings of the Cambridge Philosophical Society, LIX (1963), 285

CHAPTER III

ENUMERATION BY MACHINE

Introduction

In 1957 Coxeter and Moser stated in the introduction to their book that the "method (for systematic enumeration) is sufficiently mechanical for the use of an electronic computer."[5] Since then several people have independently written programs for various machines to accomplish this task.

Leech gave a history of the work done on this problem that was known to him at the time of publication of his paper.[6] He gave an excellent description of the work of C. B. Haselgrove and his own work. He then cited the work of R. Maddison and A. Sinkov. All of the above mentioned programs used basically the same logic and of these, the work of Sinkov is best known to the author of this paper.

Description of Logic Used by by Sinkov

The first important way in which all of the computer programs differ from the hand method is in the elimination

---

[5] H. M. S. Coxeter and W. O. T. Moser, Generators a Relations for Discrete Groups (Berlin: Springer-Verla 1957), p. v.

[6] Leech, pp. 259-263

of the tables for each relation. Instead, tables are carried only for each generator and its inverse. This results in a considerable saving of storage space and therefore permits a larger group to be enumerated. The defining relations are stored in the machine and the generators comprising the relations are fetched as needed.

Sinkov's program essentially applies each coset in turn to each of the defining relations in turn. Assume that a_0 is the current coset number, that m cosets have already been defined, and that $f_i(S_1,S_2,\ldots,S_n) = R_1R_2\ldots R_\ell = E$ is the current defining relation. Then a_0R_1 is extracted from the table for the generator R_1. If a_0R_1 is defined e.g. $a_0R_1 = a_1$, then a_1R_2 is extracted from the table and so on. If for some j, a_jR_{j+1} is not defined, it is immediately defined as coset m+1, the appropriate entries are made in the tables, and the processing continues. When the end of the relation is reached, a test is made to determine if $a_0 = a_\ell$. If $a_0 \neq a_\ell$, a coincidence has been discovered. If $a_0 = a_\ell$ or if a coincidence and all consequential coincidences have been processed, the current coset, a_0, is applied in like manner to the next relation. When the coset a_0 has been applied to all of the relations, the next coset a_0+1 is applied to the relations in turn. The process is complete when the last coset defined has been applied to all of the relations, without causing any new definitions to be introduced.

When a coincidence, $a \equiv b$, with $a < b$, is discovered, the row corresponding to coset b is examined for all generators and inverses. If a given entry, bR_i, is undefined, no action is necessary and the next entry is examined. Otherwise a test is made to see if $bR_i = b$ and if that is the case, it is replaced by a. If $bR_i \neq b$, then the inverse entry $(bR_i)R_i^{-1}$ is deleted from the table. Next the entry aR_i is examined. If aR_i is not defined, the entry bR_i is inserted. If aR_i is defined and $aR_i = b$ it is replaced by a. Otherwise a new coincidence is set up between aR_i and bR_i. Then a check is made to determine if $(aR_i)R_i^{-1}$ is defined and if not, a is inserted. Finally the entries in the row b are deleted (made zero).

The list of coincidences awaiting processing is sorted lexicographically so that redundant information need not be stored and also to assure that no coincidence is processed on a row already made zero.

After the entire list of coincidences has been processed, it is desirable for efficient use of memory space to eliminate the vacated rows from all the tables. This is easily done by using the coincidence routine to set up an artificial coincidence between the first empty row and the next non-empty one. This process is repeated until the tables are again without empty rows.

Description of Logic Used by Author

The program written by the author of this paper uses the other logic scheme presented in the literature. The method is essentially that of H. Felsch[7] although the program was written independently. This method was also used by Bandler (see Leech's paper[8]) although he did not program for automatic processing of coincidences.

The author's program was originally written for the IBM 1620, but in the spring of 1963 an IBM 7090 was delivered to the Computer Center of the University of Maryland, so the program was rewritten and modified using Fortran II for the 7090. Fortran II is a problem oriented programming language and hence a source listing of the program (see Appendix) may be of interest.

Basically the procedure used in this program is as follows: The cosets are applied sequentially to the defining relations in turn as before; however, when the forward working is halted by an undefined coset, the current coset is then applied to the inverse of the last generator in the relation. This backward working proceeds in a manner similar to the forward working and one of three things may happen. First,

---

[7] H. Felsch, "Programmierung der Restklassenabzaehlung einer Gruppe nach Untergruppen," Numerische Mathematik, III (1961), pp. 250-256

[8] Leech, p. 262.

the backward working may encounter an undefined coset in which case a new coset is defined; second, the backward working may just meet the forward working, in which case new information has been deduced; or third, the forward working and the backward working may overlap, in which case a coincidence is deduced. When a new coset is defined or when new information is deduced, e.g., $aR_i = b$, every occurrence of the generator R_i or its inverse R_i^{-1} is examined. The coset a is applied to the word in which the generator R_i appeared shifted cyclicaly to begin with R_i. The same procedure as in the general working is used, namely, upon reaching a gap in the forward working, backward working is begun. However, if a void is discovered a new definition is not made. New information discovered in this manner is entered into the multiplication tables and stored away for future processing. Coincidences are handled in the same manner as in Sinkov's program except that provision is made to make any necessary changes to the table of information awaiting processing that might have occurred due to the processing of subsequent coincidences.

In Leech's paper he stated, "No direct comparison of running times with the two methods is available at present as the machine speeds are widely different; this must wait until both methods have been programmed for the same machine."[9] A partial answer to this question is now available since Sinkov's problem was written for the IBM 704 and is capable of being run on the IBM 7090. The following chart from a report written by Sinkov for the Computer Science Center at the University of Maryland in June 1963 shows a comparison of running times on the classical problem $(8,7|2,3)$. The author's running time has been added.

TABLE I

COMPARISON OF COMPUTER RUNNING TIMES ON THE GROUP $(8,7|2,3)$

Person	Machine	Cosets Required	Time
Todd	By hand	945	>30 hours
Felsch	Zeus 22	1300	~2 hours
Leech	EDSAC 2	2000	42 minutes
Sinkov	IBM 7090	2176	5 minutes
Leech	KDF 9		2 min. 30 sec.
Snively	IBM 7090	1747	1 min. 36 sec.

The faster time obtained by the author's program is not necessarily indicative of more efficient logic. The author's logic is considerably more complicated and therefore takes up more storage space thus limiting the size of the problem that may be handled within the memory of the machine. In

[9] Leech, p. 263

fact the author's program can enumerate 10000/n cosets where n is the number of generators of the group being enumerated. Running times are expected to vary widely from problem to problem. The Felsch logic would be better in a problem that involves a large number of excess cosets to be defined since relatively few excess cosets are defined by the author's scheme.

CHAPTER IV

THE BURNSIDE PROBLEM

In 1902 Burnside stated what is commonly called the Burnside problem: can the order of a group with a finite number of generators be infinite while the period of each element in the group is finite?[10] To the author's knowledge, the problem remains unsolved, for although a Russian mathematician, Novikov, claimed to have answered it affirmatively, his proof has not yet been published.

A more specialized problem may be stated simply: assume the groups under consideration are finitely generated and that the orders of every element in the group, are bounded. Suppose, for example, $S_1, S_2, \ldots, S_r$ generate a group, $B_{n,r}$, and every element $R \in B_{n,r}$ satisfies the relation $R^n = E$. Then $B_{n,r}$ is called the Burnside group of exponent n with r generators. This Burnside problem now reduces to the question: which of the groups $B_{n,r}$ are finite?

---

[10] W. Burnside, "On an Unsettled Question in the Theory of Discontinuous Groups," Quarterly Journal of Pure and Applied Mathematics, XXXIII (1902), p. 230-238

In his book Marshall Hall summarizes the work done on this problem up to 1959.[11] Some interesting results are:

1. $B_{2,r}$ is finite (for finite r), namely the abelian group of the form:

$$\underbrace{C_2 x C_2 x \ldots x C_2}_{r \text{ factors}} \qquad (11)$$

2. $B_{3,r}$ is finite and of order:

$$3^{\frac{r(r^2+5)}{6}} \qquad (12)$$

This result was obtained by Levi and van der Waerden.[12]

3. $B_{4,r}$ is finite. This result was obtained by Sanov.[13]

4. $B_{5,2}$ if it is finite, has order at most 5^{34} (see Kostrikin)[14]

---

[11] M. Hall, The Theory of Groups. (New York: The MacMillan Company, 1959), pp. 320-338

[12] Ibid., p. 321

[13] Ibid., p. 324

[14] Ibid., p. 327

5. $B_{6,r}$ is finite and of order:

$$2^a 3^{\frac{b(b^2+5)}{6}}$$

where:

$$a = 1+(r-1)3^{\frac{r(r^2+5)}{6}} \tag{13}$$

and:

$$b = 1+(r-1)2^r$$

This result was obtained by M. Hall.[15]

Although the orders of the groups $B_{3,r}$ are known, sets of irreducible or nearly irreducible defining relations are not known for all of these groups. In attempting to find such sets of defining relations, one finds a good application for computer enumeration. The technique used is to overdefine the group, that is, to fix the periods of more elements of the group than is necessary to define the group. Then when the order of the group is thus determined (if not already known) the enumeration is rerun with some of the relations removed. If a group of the same order results, the relations removed were redundant, i.e., an algebraic consequence of the remaining relations. By proceeding in this manner, a non-redundant

---

[15] Ibid., p. 336-337

set of relations may be obtained.

$B_{3,1}$ is, of course, the cyclic group of order 3 satisfying the relation:

$$A^3 = E \qquad (14)$$

$B_{3,2}$, of order 27, satisfies the following set of relations:

$$A^3 = B^3 = (AB)^3 = (A^{-1}B)^3 = E \qquad (15)$$

In his 1963 paper, Leech gave his results for the group $B_{3,3}$.[16] He obtained his results by enumerating the 81 cosets of A,B which is $B_{3,2}$ of order 27. One of the resulting definitions is:

$$A^3 = B^3 = C^3 = (AB)^3 = (AC)^3 = (BC)^3 = (A^{-1}B)^3 =$$

$$(A^{-1}C)^3 = (B^{-1}C)^3 = (ABC)^3 = (A^{-1}BC)^3 = (AB^{-1}C)^3 =$$

$$(ABC^{-1})^3 = E$$

His enumeration was performed on EDSAC 2. The memory of EDSAC 2 was not large enough to permit the enumeration of $B_{3,4}$. This is the problem solved by the author on the IBM 7090.

In approaching the problem of $B_{3,4}$ with generators A,B,C,D the first step was to consider the generators in all combinations of three and assure that they satisfy the relations (14). When this is done, we are assured that all words containing only three of the four generators are of exponent 3.

---

[16] Leech, p. 264

To accomplish this, it was sufficient to specify the periods of 32 elements. These are:

$$A^3 = B^3 = C^3 = D^3 = E$$

$$(AB)^3 = (AC)^3 = (AD)^3 = (BC)^3 = (BD)^3 = (CD)^3 = E$$

$$(A^{-1}B)^3 = (A^{-1}C)^3 = (A^{-1}D)^3 = (B^{-1}C)^3 = (B^{-1}D)^3 = (C^{-1}D)^3 = E$$

$$(ABC)^3 = (A^{-1}BC)^3 = (AB^{-1}C)^3 = (ABC^{-1})^3 = E \tag{17}$$

$$(ACD)^3 = (A^{-1}CD)^3 = (AC^{-1}D)^3 = (ACD^{-1})^3 = E$$

$$(BCD)^3 = (B^{-1}CD)^3 = (BC^{-1}D)^3 = (BCD^{-1})^3 = E$$

Let $W_{ij} = R_1^{\xi 1}R_2^{\xi 2}R_3^{\xi 3}R_4^{\xi 4}$ where $(i = 1,2,\ldots,6)$ and $(j = 1,2,\ldots,8)$ such that if $Y_j = (\xi_1,\xi_2,\xi_3,\xi_4)$ we have

$$\begin{aligned} Y_1 &= (1,1,1,1) & Y_5 &= (1,1,1,-1) \\ Y_2 &= (-1,1,1,1) & Y_6 &= (-1,-1,1,1) \\ Y_3 &= (1,-1,1,1) & Y_7 &= (-1,1,-1,1) \\ Y_4 &= (1,1,-1,1) & Y_8 &= (-1,1,1,-1) \end{aligned} \tag{18}$$

and

$$\begin{aligned} W_{11} &= ABCD & W_{41} &= ACDB \\ W_{21} &= ABDC & W_{51} &= ACBC \\ W_{31} &= ACBD & W_{61} &= ADCB \end{aligned} \tag{19}$$

It is readily verified that if A,B,C, and D satisfy relations (17) and

$$W_{ij}^{\ 3} = E \quad (i = 1,2,\ldots,6;\ j = 1,2,\ldots,8) \tag{20}$$

that all words of four letters have period 3. A computer enumeration using relations (17) and (20) of cosets of $B_{3,3} = \{A,B,C\}$ gave a group of order 3^{14}, the known order of $B_{3,4}$. A later series of enumerations proved that it is only necessary for i to assume 3 values in (20) in order for (17) and (20) to define $B_{3,4}$ although not all choices of 3 values for i were successful. Those which were successful are:

$$\begin{array}{ll} i = 1,2,3 & i = 2,3,6 \\ i = 1,2,5 & i = 2,5,6 \\ i = 1,3,4 & i = 3,4,6 \\ i = 1,4,5 & i = 4,5,6 \end{array} \qquad (21)$$

The next experiment tried was to hold i fixed at i = 1,2,3 and vary the values permitted to j. The following sets of defining relations were thereby obtained for $B_{3,4}$:

$$\text{Relations (17) and} \quad W_{ij}^{\ 3} = E, (i=1,2,3) \left\{ \begin{array}{ll} & \text{No. of words} \\ (j = 1,2,\ldots,8) & 56 \\ (j = 2,3,\ldots,8) & 53 \\ (j = 3,4,\ldots,8) & 50 \\ (j = 4,5,\ldots,8) & 47 \\ (j = 5,6,\ldots,8) & 44 \\ (j = 6,7,8) & 41 \\ (j = 7,8) & 38 \\ (j = 1) & 35 \\ \vdots & \\ (j = 8) & 35 \end{array} \right. \qquad (22)$$

Since an enumeration of relations (17) alone failed to give closure before memory capacity was exceeded it is quite likely that the last eight successful sets of defining relations in (22) are irreducible. A complete proof of the last statement by enumerations would require 35 enumerations for each definition or about 72 hours of 7090 computer time.

CHAPTER V

OUTLOOK FOR THE FUTURE

The obvious next step would be to attempt to determine a set of defining relations for $B_{3,5}$. This, however, is a problem which far exceeds the capability of the 7090 since $B_{3,5}$ is of order 3^{25} and the largest subgroup available for an enumeration is $B_{3,4}$ of order 3^{14} which would require a total of 3^{11} cosets to be defined. Since there are tables for each generator and its inverse, a total of 10 tables, this means that a total of 1,771,470 table entries must be provided. Even packing two entries to a word ($3^{11} < 2^{18}$) only 65,536 entries can be provided, not allowing room for the program and other tables.

A means of extending the program's capabilities would be to store the tables on magnetic tape and call them into memory as needed; however, this is very impractical because such operations are quite time consuming and large amounts of computer time are not readily available.

Another possibility for solving this problem is a disk storage similar to the IBM 1301, but unfortunately this was not available to the author.

However, the work on $B_{3,4}$ did permit a conjecture. Given the definition for group $B_{3,n}$, it seems likely that in addition to the combinations of relations needed to define

$B_{3,n}$ as a subgroup of $B_{3,n+1}$ in $\frac{(n+1)n}{2}$ ways it is only necessary to add a portion of the words of n + 1 letters to completely define $B_{3,n+1}$.

Another interesting question is the study of $B_{5,2}$; however, the largest readily known subgroup available is $B_{5,1} = C_5$, the cyclic group of order 5. Even if the order of $B_{5,2}$ were as low as 5^{10} one would have to enumerate $5^9 = 1,953,125$ cosets, a task which is well beyond the capability of the 7090 without an extremely large random access storage.

In the future one may expect computers to become faster and to have larger memories. At the current machine speed a memory of 5,000,000 IBM 7090 words would enable the author's program to undertake the problem of $B_{3,5}$ and possibly $B_{5,2}$, however, the time required to run these problems would be prohibitive. With a hundred fold increase in processing speed these two problems would be well within the range of machine enumeration.

APPENDIX

SOURCE LISTING OF

COSET ENUMERATION PROGRAM

```
*     LIST8
*     LABEL
C     7090 COSET EVALUATION - FELSCH LOGIC - JAMES W. SNIVELY, JR.
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100)
      DIMENSION INFO1(50),INFO2(50),INFO3(50)
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT
      NJOB=1
      IN=5
      NOUT=6
      LPWMAX=14
      LWPGMX=200
      ICTMAX=20000
      IWMAX=50
      ICMAX=1000
      IWAIT=0
      READ INPUT TAPE IN,60,NJOBS
   10 CALL CLOCK(6H= TIME,-1)
      CALL INPUT
      CALL ARRAY
      CALL RESET
      CALL SCAN
      IF(NJOB-NJOBS) 40,50,50
   40 NJOB=NJOB+1
      GO TO 10
   50 CALL CLOCK(6H= TIME,-1)
      CALL EXIT
   60 FORMAT(I5)
      END

*     LIST8
*     LABEL
      SUBROUTINE INPUT
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100)
      DIMENSION INFO1(50),INFO2(50),INFO3(50)
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT
      READ INPUT TAPE IN,20,NWORD
      DO 10 I=1,NWORD
      READ INPUT TAPE IN,30,NLETR(I),IPOWR(I)
      K=NLETR(I)
   10 READ INPUT TAPE IN,40,(IGEN(I,J),J=1,K)
      READ INPUT TAPE IN,50,NORD,ITR,NGENS
      RETURN
   20 FORMAT(I5)
   30 FORMAT(2I5)
   40 FORMAT(13I5)
   50 FORMAT(3I5)
      END
```

```
                                                                        ARRAY
*     LIST8                                                             ARRAY 01
*     LABEL                                                             ARRAY 02
      SUBROUTINE ARRAY                                                  ARRAY 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     ARRAY 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) ARRAY 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           ARRAY 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        ARRAY 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    ARRAY 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             ARRAY 09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       ARRAY 10
      DO 40 I=1,NGENS                                                   ARRAY 11
      II=1                                                              ARRAY 12
      DO 30 J=1,NWORD                                                   ARRAY 13
      JJ=NLETR(J)                                                       ARRAY 14
      DO 30 K=1,JJ                                                      ARRAY 15
      IF(XABSF(IGEN(J,K))-I) 30,20,30                                   ARRAY 16
   20 ITBL(I,II)=LPWMAX*J+K                                             ARRAY 17
      II=II+1                                                           ARRAY 18
      IF(II-LWPGMX) 30,30,50                                            ARRAY 19
   30 CONTINUE                                                          ARRAY 20
   40 IWPG(I)=II-1                                                      ARRAY 21
      RETURN                                                            ARRAY 22
   50 CALL ERROR(1)                                                     ARRAY 23
      RETURN                                                            ARRAY 24
      END                                                               ARRAY 25
                                                                        RESET
*     LIST8                                                             RESET 01
*     LABEL                                                             RESET 02
      SUBROUTINE RESET                                                  RESET 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     RESET 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) RESET 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           RESET 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        RESET 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    RESET 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             RESET 09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       RESET 10
      MAXROW=ICTMAX/(2*NGENS)                                           RESET 11
      N=2*NGENS*MAXROW                                                  RESET 12
      DO 10 I=1,N                                                       RESET 13
   10 ICHRT(I)=0                                                        RESET 14
      READ INPUT TAPE IN,60,N,NDEF                                      RESET 15
      IF(N) 40,40,20                                                    RESET 16
   20 DO 30 I=1,N                                                       RESET 17
      READ INPUT TAPE IN,70,J,K,L                                       RESET 18
      CALL SEEK(-J,K,M)                                                 RESET 19
      ICHRT(M)=L                                                        RESET 20
      CALL SEEK(-L,-K,M)                                                RESET 21
   30 ICHRT(M)=J                                                        RESET 22
   40 MAX=NDEF                                                          RESET 23
      IR=0                                                              RESET 24
      DO 50 I=1,NWORD                                                   RESET 25
   50 ITALLY(I)=0                                                       RESET 26
      RETURN                                                            RESET 27
```

```
   60 FORMAT(2I5)                                                        RESET 28
   70 FORMAT(3I5)                                                        RESET 29
      END                                                                RESET 30
                                                                         SCAN
*     LIST8                                                              SCAN  01
*     LABEL                                                              SCAN  02
      SUBROUTINE SCAN                                                    SCAN  03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)      SCAN  04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100)  SCAN  05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                            SCAN  06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR         SCAN  07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS     SCAN  08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL              SCAN  09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                        SCAN  10
      GO TO (3000,1000,1000,1000),ITR                                    SCAN  11
 1000 WRITE OUTPUT TAPE NOUT,2000                                        SCAN  12
 2000 FORMAT(5H1SCAN)                                                    SCAN  13
 3000 ISCAN=0                                                            SCAN  14
   10 ISCAN=ISCAN+1                                                      SCAN  15
      DO 150 J=1,NWORD                                                   SCAN  16
   15 IWORK=ISCAN                                                        SCAN  17
      KK=IPOWR(J)                                                        SCAN  18
      LL=NLETR(J)                                                        SCAN  19
      DO 20 K=1,KK                                                       SCAN  20
      DO 20 L=1,LL                                                       SCAN  21
      CALL SEEK(IWORK,IGEN(J,L),M)                                       SCAN  22
      GO TO (20,40),M                                                    SCAN  23
   20 CONTINUE                                                           SCAN  24
      IF(IWORK-ISCAN) 30,150,30                                          SCAN  25
   30 CALL COINC(IWORK,ISCAN,J)                                          SCAN  26
      GO TO 145                                                          SCAN  27
   40 IST=IWORK                                                          SCAN  28
      IWORK=ISCAN                                                        SCAN  29
      KT=K+1                                                             SCAN  30
      LT=L+1                                                             SCAN  31
      IF(KT-KK) 50,50,70                                                 SCAN  32
   50 DO 60 K=KT,KK                                                      SCAN  33
      DO 60 N=1,LL                                                       SCAN  34
      L=(LL+1)-N                                                         SCAN  35
      CALL SEEK(IWORK,-IGEN(J,L),M)                                      SCAN  36
      GO TO (60,140),M                                                   SCAN  37
   60 CONTINUE                                                           SCAN  38
   70 IF(LT-LL) 80,80,100                                                SCAN  39
   80 DO 90 N=LT,LL                                                      SCAN  40
      L=(LL+LT)-N                                                        SCAN  41
      CALL SEEK(IWORK,-IGEN(J,L),M)                                      SCAN  42
      GO TO (90,140),M                                                   SCAN  43
   90 CONTINUE                                                           SCAN  44
  100 L=LT-1                                                             SCAN  45
      CALL SEEK(IWORK,-IGEN(J,L),M)                                      SCAN  46
      GO TO (110,120),M                                                  SCAN  47
  110 CALL COINC(IWORK,IST,J)                                            SCAN  48
      GO TO 145                                                          SCAN  49
 -120 CALL INFO(IST,IGEN(J,L),IWORK)                                     SCAN  50
```

```
      GO TO 130                                                         SCAN  51
  130 CALL LOOKI(IST,IGEN(J,L),IWORK)                                   SCAN  52
      GO TO 150                                                         SCAN  53
  140 L=LT-1                                                            SCAN  54
      CALL DEFINE(IST,IGEN(J,L))                                        SCAN  55
      GO TO 15                                                          SCAN  56
  145 CALL CLOSE                                                        SCAN  57
  150 CONTINUE                                                          SCAN  58
      IF(ISCAN-NDEF) 10,160,160                                         SCAN  59
  160 CALL FINISH                                                       SCAN  60
      RETURN                                                            SCAN  61
      END                                                               SCAN  62
                                                                        DEFINE
*     LIST8                                                             DEFINE01
*     LABEL                                                             DEFINE02
      SUBROUTINE DEFINE(INX,INY)                                        DEFINE03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     DEFINE04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) DEFINE05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           DEFINE06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        DEFINE07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    DEFINE08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             DEFINE09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       DEFINE10
      GO TO (3000,1000,1000,1000),ITR                                   DEFINE11
 1000 WRITE OUTPUT TAPE NOUT,2000,INX,INY                               DEFINE12
 2000 FORMAT(8H DEFINE(,I5,1H,,I5,1H))                                  DEFINE13
 3000 IX=INX                                                            DEFINE14
      IY=INY                                                            DEFINE15
      NDEF=NDEF+1                                                       DEFINE16
      IF(NDEF-MAX) 20,20,10                                             DEFINE17
   10 MAX=NDEF                                                          DEFINE18
   20 IF(NDEF-MAXROW) 40,40,30                                          DEFINE19
   30 CALL ERROR(2)                                                     DEFINE20
   40 CALL SEEK(-IX,IY,I)                                               DEFINE21
      ICHRT(I)=NDEF                                                     DEFINE22
      CALL SEEK(-NDEF,-IY,I)                                            DEFINE23
      ICHRT(I)=IX                                                       DEFINE24
      CALL LOOKI(IX,IY,NDEF)                                            DEFINE25
      RETURN                                                            DEFINE26
      END                                                               DEFINE27
                                                                        LOOKI
*     LIST8                                                             LOOKI 01
*     LABEL                                                             LOOKI 02
      SUBROUTINE LOOKI(INX,INY,INZ)                                     LOOKI 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     LOOKI 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) LOOKI 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           LOOKI 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        LOOKI 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    LOOKI 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             LOOKI 09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       LOOKI 10
      GO TO (3000,3000,3000,1000),ITR                                   LOOKI 11
 1000 WRITE OUTPUT TAPE NOUT,2000,INX,INY,INZ                           LOOKI 12
 2000 FORMAT(7H LOOKI(,I5,1H,,I5,1H,,I5,1H))                            LOOKI 13
```

```
 3000 IX=INX                                                            LOOKI 14
      IY=INY                                                            LOOKI 15
      IZ=INZ                                                            LOOKI 16
      IWAIT=0                                                           LOOKI 17
      CALL LOOK(IX,IY,IZ)                                               LOOKI 18
      IF(IWAIT) 40,40,20                                                LOOKI 19
   20 I=IWAIT                                                           LOOKI 20
      IWAIT=I-1                                                         LOOKI 21
      CALL LOOK(INFO1(I),INFO2(I),INFO3(I))                             LOOKI 22
      IF(IWAIT) 40,40,20                                                LOOKI 23
   40 RETURN                                                            LOOKI 24
      END                                                               LOOKI 25
                                                                        LOOK
*     LIST8                                                             LOOK  01
*     LABEL                                                             LOOK  02
      SUBROUTINE LOOK(INX,INY,INZ)                                      LOOK  03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     LOOK  04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) LOOK  05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           LOOK  06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        LOOK  07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    LOOK  08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             LOOK  09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       LOOK  10
      GO TO (3000,3000,3000,1000),ITR                                   LOOK  11
 1000 WRITE OUTPUT TAPE NOUT,2000,INX,INY,INZ                           LOOK  12
 2000 FORMAT(6H LOOK(,I5,1H,,I5,1H,,I5,1H))                             LOOK  13
 3000 IF(INY) 10,20,20                                                  LOOK  14
   10 ILOOK=INZ                                                         LOOK  15
      IZ=INX                                                            LOOK  16
      IY=-INY                                                           LOOK  17
      GO TO 30                                                          LOOK  18
   20 ILOOK=INX                                                         LOOK  19
      IZ=INZ                                                            LOOK  20
      IY=INY                                                            LOOK  21
   30 II=IWPG(IY)                                                       LOOK  22
      DO 230 I=1,II                                                     LOOK  23
      IWORK=ILOOK                                                       LOOK  24
      LL=ITBL(IY,I)/LPWMAX                                              LOOK  25
      KL=ITBL(IY,I)-LPWMAX*LL                                           LOOK  26
      JJ=IPOWR(LL)                                                      LOOK  27
      KK=NLETR(LL)                                                      LOOK  28
      DO 60 J=1,JJ                                                      LOOK  29
      DO 60 M=1,KK                                                      LOOK  30
      K=M+KL-1                                                          LOOK  31
      IF(K-KK) 50,50,40                                                 LOOK  32
   40 K=K-KK                                                            LOOK  33
   50 CALL SEEK(IWORK,IGEN(LL,K),L)                                     LOOK  34
      GO TO (60,80),L                                                   LOOK  35
   60 CONTINUE                                                          LOOK  36
      IF(IWORK-ILOOK) 70,230,70                                         LOOK  37
   70 CALL COINC(IWORK,ILOOK,LL)                                        LOOK  38
      GO TO 225                                                         LOOK  39
   80 IST=IWORK                                                         LOOK  40
      IWORK=ILOOK                                                       LOOK  41
```

```
      JT=J+1                                                            LOOK  42
      KT=M+1                                                            LOOK  43
      IF(JT-JJ) 90,90,130                                               LOOK  44
   90 DO 120 J=JT,JJ                                                    LOOK  45
      DO 120 M=1,KK                                                     LOOK  46
      K=KK+KL-M                                                         LOOK  47
      IF(K-KK) 110,110,100                                              LOOK  48
  100 K=K-KK                                                            LOOK  49
  110 CALL SEEK(IWORK,-IGEN(LL,K),L)                                    LOOK  50
      GO TO (120,230),L                                                 LOOK  51
  120 CONTINUE                                                          LOOK  52
  130 IF(KT-KK) 140,140,180                                             LOOK  53
  140 DO 170 M=KT,KK                                                    LOOK  54
      K=(KK+KL+KT)-(M+1)                                                LOOK  55
      IF(K-KK) 160,160,150                                              LOOK  56
  150 K=K-KK                                                            LOOK  57
  160 CALL SEEK(IWORK,-IGEN(LL,K),L)                                    LOOK  58
      GO TO (170,230),L                                                 LOOK  59
  170 CONTINUE                                                          LOOK  60
  180 K=KL+KT-2                                                         LOOK  61
      IF(K-KK) 200,200,190                                              LOOK  62
  190 K=K-KK                                                            LOOK  63
  200 CALL SEEK(IWORK,-IGEN(LL,K),L)                                    LOOK  64
      GO TO (210,220),L                                                 LOOK  65
  210 CALL COINC(IWORK,IST,LL)                                          LOOK  66
      GO TO 225                                                         LOOK  67
  220 CALL INFO(IST,IGEN(LL,K),IWORK)                                   LOOK  68
      GO TO 230                                                         LOOK  69
  225 CALL CLOSE                                                        LOOK  70
  230 CONTINUE                                                          LOOK  71
      RETURN                                                            LOOK  72
      END                                                               LOOK  73
                                                                        INFO
*     LIST8                                                             INFO  01
*     LABEL                                                             INFO  02
      SUBROUTINE INFO(IX,IY,IZ)                                         INFO  03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     INFO  04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) INFO  05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           INFO  06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        INFO  07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    INFO  08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             INFO  09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       INFO  10
      GO TO (3000,1000,1000,1000),ITR                                   INFO  11
 1000 WRITE OUTPUT TAPE NOUT,2000,IX,IY,IZ                              INFO  12
 2000 FORMAT(6H INFO(,I5,1H,,I5,1H,,I5,1H))                             INFO  13
 3000 CALL SEEK(-IX,IY,L)                                               INFO  14
      ICHRT(L)=IZ                                                       INFO  15
      CALL SEEK(-IZ,-IY,L)                                              INFO  16
      ICHRT(L)=IX                                                       INFO  17
      IWAIT=IWAIT+1                                                     INFO  18
      IF(IWAIT-IWMAX) 10,10,20                                          INFO  19
   10 INFO1(IWAIT)=IX                                                   INFO  20
      INFO2(IWAIT)=IY                                                   INFO  21
```

```
      INFO3(IWAIT)=IZ                                                   INFO  22
      RETURN                                                            INFO  23
   20 CALL ERROR(3)                                                     INFO  24
      RETURN                                                            INFO  25
      END                                                               INFO  26
                                                                        SEEK
*     LIST8                                                             SEEK  01
*     LABEL                                                             SEEK  02
      SUBROUTINE SEEK(IX,IY,IZ)                                         SEEK  03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     SEEK  04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) SEEK  05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           SEEK  06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        SEEK  07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    SEEK  08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             SEEK  09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       SEEK  10
      XLCTF(I,J)=2*NGENS*(I-1)+2*(XABSF(J)-1)+(3-XSIGNF(1,J))/2         SEEK  11
      GO TO (3000,3000,3000,1000),ITR                                   SEEK  12
 1000 WRITE OUTPUT TAPE NOUT,2000,IX,IY,IZ                              SEEK  13
 2000 FORMAT(6H SEEK(,I5,1H,,I5,1H,,I5,1H))                             SEEK  14
 3000 IF(IX) 10,20,20                                                   SEEK  15
   10 IZ=XLCTF(-IX,IY)                                                  SEEK  16
      RETURN                                                            SEEK  17
   20 IZ=XLCTF(IX,IY)                                                   SEEK  18
      IF(ICHRT(IZ)) 30,40,30                                            SEEK  19
   30 IX=ICHRT(IZ)                                                      SEEK  20
      IZ=1                                                              SEEK  21
      RETURN                                                            SEEK  22
   40 IZ=2                                                              SEEK  23
      RETURN                                                            SEEK  24
      END                                                               SEEK  25
                                                                        COINC
*     LIST8                                                             COINC 01
*     LABEL                                                             COINC 02
      SUBROUTINE COINC(INX,INY,INK)                                     COINC 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     COINC 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) COINC 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           COINC 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        COINC 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    COINC 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             COINC 09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       COINC 10
      IF(INK) 25,25,10                                                  COINC 11
   10 GO TO (20,1000,1000,1000),ITR                                     COINC 12
 1000 WRITE OUTPUT TAPE NOUT,2000,INX,INY,INK                           COINC 13
 2000 FORMAT(7H COINC(,I5,1H,,I5,1H,,I5,1H))                            COINC 14
   20 ITALLY(INK)=ITALLY(INK)+1                                         COINC 15
      N=0                                                               COINC 16
      GO TO 30                                                          COINC 17
   25 N=-1                                                              COINC 18
   30 CALL ENTER(INX,INY,N)                                             COINC 19
   40 IX=ICOIN1(1)                                                      COINC 20
      IY=ICOIN2(1)                                                      COINC 21
      IF(ILOOK-IY) 60,50,60                                             COINC 22
```

```
   50 ILOOK=IX                                                          COINC 23
   60 IF(IWAIT) 120,120,70                                              COINC 24
   70 DO 110 I=1,IWAIT                                                  COINC 25
      IF(INFO1(I)-IY) 90,80,90                                          COINC 26
   80 INFO1(I)=IX                                                       COINC 27
   90 IF(INFO3(I)-IY) 110,100,110                                       COINC 28
  100 INFO3(I)=IX                                                       COINC 29
  110 CONTINUE                                                          COINC 30
  120 CALL SEEK(-IX,1,II)                                               COINC 31
      CALL SEEK(-IY,1,JJ)                                               COINC 32
      DO 210 K=1,NGENS                                                  COINC 33
      DO 210 L=1,2                                                      COINC 34
      M=2*K+L-3                                                         COINC 35
      I=II+M                                                            COINC 36
      J=JJ+M                                                            COINC 37
      IF(ICHRT(J)) 130,210,130                                          COINC 38
  130 IF(ICHRT(J)-IY) 150,140,150                                       COINC 39
  140 ICHRT(J)=IX                                                       COINC 40
      GO TO 155                                                         COINC 41
  150 CALL SEEK(-ICHRT(J),XSIGNF(K,2*L-3),M)                            COINC 42
      ICHRT(M)=0                                                        COINC 43
  155 IF(ICHRT(I)) 160,190,160                                          COINC 44
  160 IF(ICHRT(I)-IY) 180,170,180                                       COINC 45
  170 ICHRT(I)=IX                                                       COINC 46
  180 CALL ENTER(ICHRT(I),ICHRT(J),N)                                   COINC 47
      GO TO 200                                                         COINC 48
  190 ICHRT(I)=ICHRT(J)                                                 COINC 49
  200 CALL SEEK(-ICHRT(I),XSIGNF(K,2*L-3),M)                            COINC 50
      IF(ICHRT(M)) 206,203,206                                          COINC 51
  203 ICHRT(M)=IX                                                       COINC 52
  206 ICHRT(J)=0                                                        COINC 53
  210 CONTINUE                                                          COINC 54
      IF(INK) 250,250,220                                               COINC 55
  220 IF(N-1) 250,250,230                                               COINC 56
  230 N=N-1                                                             COINC 57
      DO 240 I=1,N                                                      COINC 58
      ICOIN1(I)=ICOIN1(I+1)                                             COINC 59
  240 ICOIN2(I)=ICOIN2(I+1)                                             COINC 60
      ICOIN1(N+1)=0                                                     COINC 61
      ICOIN2(N+1)=0                                                     COINC 62
      GO TO 40                                                          COINC 63
  250 N=0                                                               COINC 64
      RETURN                                                            COINC 65
      END                                                               COINC 66
                                                                        ENTER
*     LIST8                                                             ENTER 01
*     LABEL                                                             ENTER 02
      SUBROUTINE ENTER(INX,INY,N)                                       ENTER 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     ENTER 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) ENTER 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           ENTER 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        ENTER 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    ENTER 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             ENTER 09
```

```
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT
      IF(INX-INY) 10,170,20
   10 IX=INX
      IY=INY
      GO TO 30
   20 IX=INY
      IY=INX
   30 IF(N) 240,220,40
   40 IF(IY-ICOIN2(1)) 100,50,80
   50 IF(IX-ICOIN1(1)) 70,170,60
   60 IY=IX
      IX=ICOIN1(1)
      GO TO 100
   70 IY=ICOIN1(1)
      GO TO 100
   80 IF(IX-ICOIN2(1)) 100,90,100
   90 IX=ICOIN1(1)
  100 IF(N-1) 220,220,110
  110 DO 150 I=2,N
      IF(ICOIN2(I)-IY) 180,120,150
  120 IF(ICOIN1(I)-IX) 130,170,140
  130 IY=IX
      IX=ICOIN1(I)
      ICOIN1(I)=IY
      GO TO 110
  140 ICOIN1(I)=IX
      GO TO 110
  150 CONTINUE
      IF(N-ICMAX) 220,160,160
  160 CALL ERROR(4)
  170 RETURN
  180 IF(N-ICMAX) 190,160,160
  190 IF(N-IR) 196,196,193
  193 IR=N
  196 DO 200 J=I,N
      K=I+N-J
      ICOIN1(K+1)=ICOIN1(K)
  200 ICOIN2(K+1)=ICOIN2(K)
      N=N+1
      ICOIN1(I)=IX
      ICOIN2(I)=IY
  210 GO TO (170,170,1000,1000),ITR
 1000 WRITE OUTPUT TAPE NOUT,2000,IX,IY,N
 2000 FORMAT(7H ENTER(,I5,1H,,I5,1H,,I5,1H))
      GO TO 170
  220 IF(N-ICMAX) 223,160,160
  223 IF(N-IR) 230,230,226
  226 IR=N
  230 N=N+1
      ICOIN1(N)=IX
      ICOIN2(N)=IY
      GO TO 210
  240 ICOIN1(1)=IX
      ICOIN2(1)=IY
```

```
      N=1                                                               ENTER 64
      GO TO 170                                                         ENTER 65
      END                                                               ENTER 66
                                                                        CLOSE
*     LIST8                                                             CLOSE 01
*     LABEL                                                             CLOSE 02
      SUBROUTINE CLOSE                                                  CLOSE 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     CLOSE 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) CLOSE 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           CLOSE 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        CLOSE 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    CLOSE 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             CLOSE 09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       CLOSE 10
      L=2*NGENS                                                         CLOSE 11
      J=0                                                               CLOSE 12
      M=0                                                               CLOSE 13
   10 J=J+1                                                             CLOSE 14
      I2=L*J                                                            CLOSE 15
      I1=I2-L+1                                                         CLOSE 16
      DO 20 I=I1,I2                                                     CLOSE 17
      IF(ICHRT(I)) 30,20,30                                             CLOSE 18
   20 CONTINUE                                                          CLOSE 19
      M=M+1                                                             CLOSE 20
      GO TO 50                                                          CLOSE 21
   30 IF(M) 40,50,40                                                    CLOSE 22
   40 CALL COINC(J,J-M,0)                                               CLOSE 23
   50 IF(ISCAN-J) 70,60,70                                              CLOSE 24
   60 ISCAN=J-M                                                         CLOSE 25
   70 IF(J-NDEF) 10,80,80                                               CLOSE 26
   80 NDEF=J-M                                                          CLOSE 27
      RETURN                                                            CLOSE 28
      END                                                               CLOSE 29
                                                                        FINISH
*     LIST8                                                             FINISH01
*     LABEL                                                             FINISH02
      SUBROUTINE FINISH                                                 FINISH03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     FINISH04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) FINISH05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           FINISH06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        FINISH07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    FINISH08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             FINISH09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       FINISH10
      NORD=NDEF*NORD                                                    FINISH11
      WRITE OUTPUT TAPE NOUT,30,NORD,MAX,IR                             FINISH12
      WRITE OUTPUT TAPE NOUT,60                                         FINISH13
      DO 9 I=1,NWORD                                                    FINISH14
    9 WRITE OUTPUT TAPE NOUT,70,I,ITALLY(I)                             FINISH15
      DO 10 I=1,NGENS                                                   FINISH16
      INFO1(I)=I                                                        FINISH17
   10 INFO3(I)=-I                                                       FINISH18
      WRITE OUTPUT TAPE NOUT,40,(INFO1(I),INFO3(I),I=1,NGENS)           FINISH19
      DO 20 I=1,NDEF                                                    FINISH20
```

```
      J=2*(I-1)*NGENS+1                                                 FINISH21
      K=2*I*NGENS                                                       FINISH22
   20 WRITE OUTPUT TAPE NOUT,50,I,(ICHRT(L),L=J,K)                      FINISH23
      RETURN                                                            FINISH24
   30 FORMAT(26H1THE ORDER OF THE GROUP IS,I8,5H MAX=,I8,7H COINC=,I5/) FINISH25
   40 FORMAT(6X,18I6/)                                                  FINISH26
   50 FORMAT(19I6)                                                      FINISH27
   60 FORMAT(10X,10HWORD TALLY/)                                        FINISH28
   70 FORMAT(8X,2I5)                                                    FINISH29
      END                                                               FINISH30
                                                                        ERROR
*     LIST8                                                             ERROR 01
*     LABEL                                                             ERROR 02
      SUBROUTINE ERROR(I)                                               ERROR 03
      DIMENSION ICHRT(20000),ICOIN1(1000),ICOIN2(1000),IGEN(100,13)     ERROR 04
      DIMENSION IPOWR(100),NLETR(100),ITBL(10,200),IWPG(10),ITALLY(100) ERROR 05
      DIMENSION INFO1(50),INFO2(50),INFO3(50)                           ERROR 06
      COMMON NWORD,NORD,NUM,NGENS,MAX,MAXROW,NDEF,IWORK,IWAIT,IR        ERROR 07
      COMMON LPWMAX,LWPGMX,ICTMAX,IWMAX,ICMAX,IN,NOUT,ITR,NJOB,NJOBS    ERROR 08
      COMMON ISCAN,ILOOK,ITALLY,INFO1,INFO2,INFO3,IWPG,ITBL             ERROR 09
      COMMON NLETR,IPOWR,IGEN,ICOIN1,ICOIN2,ICHRT                       ERROR 10
      GO TO (10,30,50,70),I                                             ERROR 11
   10 WRITE OUTPUT TAPE NOUT,20                                         ERROR 12
      GO TO 90                                                          ERROR 13
   20 FORMAT(34H1MAXIMUM WORDS/GENERATOR EXCEEDED.//)                   ERROR 14
   30 WRITE OUTPUT TAPE NOUT,40                                         ERROR 15
      GO TO 90                                                          ERROR 16
   40 FORMAT(39H1MULTIPLICATION TABLE STORAGE EXCEEDED.//)              ERROR 17
   50 WRITE OUTPUT TAPE NOUT,60                                         ERROR 18
      GO TO 90                                                          ERROR 19
   60 FORMAT(44H1WAITING INFORMATION TABLE STORAGE EXCEEDED.//)         ERROR 20
   70 WRITE OUTPUT TAPE NOUT,80                                         ERROR 21
      GO TO 90                                                          ERROR 22
   80 FORMAT(37H1COINCIDENCE TABLE CAPACITY EXCEEDED.//)                ERROR 23
   90 CALL CLOCK(6H ERROR,-1)                                           ERROR 24
      CALL DUMP(ICHRT(20000),ICHRT(19800),2,ICHRT(200),NWORD,2)         ERROR 25
      RETURN                                                            ERROR 26
      END                                                               ERROR 27
                                                                        00000580
```

LIST OF REFERENCES

(1) Burnside, W. "On an Unsettled Question in the Theory of Discontinuous Groups." Quarterly Journal of Pure and Applied Mathematics, 33 (1902), 230-238.

(2) Coxeter, H. S. M. "The Abstract Group $G^{3,7,16}$." Proceedings of the Edinburgh Mathematical Society, 13, Series II (June 1962), 47-61.

(3) Coxeter, H. S. M. and Moser, W. O. J. Generators and Relations for Discrete Groups (Springer-Verlag, Berlin, 1957).

(4) Coxeter, H. S. M. and Todd, J. A. "A Practical Method for Enumerating Cosets of a Finite Abstract Group." Proceedings of the Edinburgh Mathematical Society 5, Series II (1936), 26-34.

(5) Felsch, H. "Programmierung der Restklassenabzaehlung einer Gruppe nach Untergruppen." Numerische Mathematik, 3 (1961), 250-256.

(6) Hall, M. The Theory of Groups (MacMillan, New York, 1959).

(7) Leech, J. "Coset Enumeration on Digital Computers." Proceedings of the Cambridge Philosophical Society, 59 (1963), 257-267.